

**WEST**[Help](#)[Logout](#)[Interrupt](#)[Main Menu](#)[Search Form](#)[Posting Counts](#)[Show S Numbers](#)[Edit S Numbers](#)[Preferences](#)[Cases](#)**Search Results -**

Term	Documents
META	49280
METAS	113
DATA	2568186
DATUM	23790
I	3221298
IS	339890
NODE	204365
NODES	125109
INODE	353
INODES	184
(3 SAME ((INODE OR (META ADJ DATA)) OR (I ADJ NODE))).USPT,PGPB,JPAB,EPAB,DWPI,TDBD.	10
(L3 SAME ((META ADJ DATA) OR (I ADJ NODE) OR INODE)).USPT,PGPB,JPAB,EPAB,DWPI,TDBD.	10

**Database:**

US Patents Full-Text Database  
US Pre-Grant Publication Full-Text Database  
JPO Abstracts Database  
EPO Abstracts Database  
Derwent World Patents Index  
IBM Technical Disclosure Bulletins

**Search:**

L8

[Refine Search](#)[Recall Text](#)[Clear](#)**Search History****DATE:** Saturday, July 12, 2003   [Printable Copy](#)   [Create Case](#)



<u>Set Name</u>	<u>Query</u>	<u>Hit Count</u>	<u>Set Name</u>
side by side			result set
<i>DB=USPT,PGPB,JPAB,EPAB,DWPI,TDBD; PLUR=YES; OP=ADJ</i>			
<u>L8</u>	l3 same ((meta adj data) or (i adj node) or inode)	10	<u>L8</u>
<u>L7</u>	parallel file same l3	21	<u>L7</u>
<u>L6</u>	l2 same l3	2	<u>L6</u>
<u>L5</u>	l2 and l3	42	<u>L5</u>
<u>L4</u>	l3 and l2 and l1	16	<u>L4</u>
<u>L3</u>	(prefetch or prefetching)	6622	<u>L3</u>
<u>L2</u>	inode	390	<u>L2</u>
<u>L1</u>	parallel adj3 file adj3 system	154	<u>L1</u>

END OF SEARCH HISTORY



Find: 

Documents

Citations

Searching for **inode and (prefetch or prefetching)**.Restrict to: [Header](#) [Title](#) Order by: [Citations](#) [Hubs](#) [Usage](#) [Date](#) Try: [Amazon](#) [B&N](#) [Google \(RI\)](#) [Google \(Web\)](#) [CSB](#) [DBLP](#)15 documents found. **Order: citations weighted by year.**[Embedded Inodes and Explicit Grouping: Exploiting Disk.. - Ganger, Kaashoek \(1997\) \(Correct\) \(14 citations\)](#)[Embedded \*\*Inodes\*\* and Explicit Grouping: Exploiting Disk Bandwidth](#)[www.sois.alaska.edu/CSLG\\_index/usenix\\_tech.97/PROCEEDINGS/ganger.ps](http://www.sois.alaska.edu/CSLG_index/usenix_tech.97/PROCEEDINGS/ganger.ps)[Implementation and Evaluation of a Multimedia File System - Niranjan Transarc \(1997\) \(Correct\) \(9 citations\)](#)

UFS 1 associates each file with a unique **inode** whose contents constitute the file metadata. This playback performance by supporting intelligent **prefetching** and state-based caching, prioritized playback performance by supporting intelligent **prefetching** and state-based caching, prioritized real-time

[www.ecsl.cs.sunysb.edu/tr/TR30.ps.Z](http://www.ecsl.cs.sunysb.edu/tr/TR30.ps.Z)

[Fast Concurrent Dynamic Linking for an Adaptive Operating System - Crispin Cowan \(1996\) \(Correct\) \(13 citations\)](#)

file type, the file system block size, whether the **inode** is in memory, and if so, its address, etc. Thus, to the application, e.g. a specialized file **prefetching** policy. Quasi-invariants can be falsified, System, Such As Paging Policy, Or File System **Prefetching** Policy. 4 The Replugging Facility The

[cse.ogi.edu/pub/dsrg/synhetix/iccds96.ps.gz](http://cse.ogi.edu/pub/dsrg/synhetix/iccds96.ps.gz)

[Efficient Support for Incremental Customization of OS Services - Peter Druschel \(Correct\) \(9 citations\)](#)

the file system using an external pager, the **inode** pager. Consider our previous example of in functionality. For example, to modify the **prefetch** strategy of a file system, it should be to provide a new function that implements the **prefetch** policy, not to re-implement the entire file

[ftp.cs.arizona.edu/xkernel/Papers/customization.ps](http://ftp.cs.arizona.edu/xkernel/Papers/customization.ps)

[LIBTP: Portable, Modular Transactions for UNIX - Seltzer, Olson \(1992\) \(Correct\) \(4 citations\)](#)

face of crashes. When a new file is created, its **inode** is written to disk before the new file is added multiple processes can access the shared data, **prefetching** may be accomplished by separate processes or processes or threads whose sole purpose is to **prefetch** pages and wait on them. There is still no way to

[wuarhive.wustl.edu/packages/postgres/papers/ERL-M92-02.ps.Z](http://wuarhive.wustl.edu/packages/postgres/papers/ERL-M92-02.ps.Z)

[A Class-Based Disk Scheduling Algorithm: Implementation and.. - Bennett, Melski \(1994\) \(Correct\) \(1 citation\)](#)

3 In Linux, all writes that maintain directory and **inode** structures are timecritical. Time-limited: A Results of this work include scheduling, **prefetching**, and caching algorithms. Carson92] Results of this work include scheduling, **prefetching**, and caching algorithms. Carson92]

[www.cs.wisc.edu/~sbennett/class\\_papers/os\\_paper.ps](http://www.cs.wisc.edu/~sbennett/class_papers/os_paper.ps)

[Damelo! An Explicitly Co-locating Web Cache File System - Jonathan Ledlie The \(Correct\)](#)

co-location works more and more poorly over time as **inode** references become scattered over distant file. Fetching one of these objects leads to a **prefetch** for all. One system further improves performance frequently accessed together nearby, they hope to **prefetch** some objects and to have their meta-data cached

[www.eecs.harvard.edu/~jonathan/wisc/mastersthesis/damelo.pdf](http://www.eecs.harvard.edu/~jonathan/wisc/mastersthesis/damelo.pdf)

[The Utility of File Names - Ellard, Ledlie, Seltzer \(2003\) \(Correct\)](#)

variants of FFS such as C-FFS [5] which co-locates **inode** and directory information to improve directory requests, where there has been extensive work in **prefetching** (sometimes called pre-caching) but it is requests, where there has been extensive work in **prefetching** (sometimes called pre-caching) but it is also

[www.eecs.harvard.edu/sos/techs/tr-05-03.ps](http://www.eecs.harvard.edu/sos/techs/tr-05-03.ps)

[Prefetching and Caching Metadata in a Distributed NFS Server - Wei, Liu, Ou, Zheng, Wu.. \(2000\) \(Correct\)](#)

Hash Chain, Free Chain, LRU Chain, Key and **Inode**. The Free Chain and LRU Chain consist of the free **Prefetching** And Caching Metadata In A Distributed Nfs of the distributed server and employs the **prefetching** approach to optimize the performance of

[chooyu.cs.uiuc.edu/~liuwei/papers/prefetch.ps](http://chooyu.cs.uiuc.edu/~liuwei/papers/prefetch.ps)



Semantically-Smart Disk Systems - Sivathanu, Prabhakaran, Popovici.. (2003) (Correct)  
system (e.g. bitmaps for tracking free space, **inodes**, data blocks, directories, and indirect blocks)  
[www.cs.wisc.edu/wind/Publications/sds-fast03.ps](http://www.cs.wisc.edu/wind/Publications/sds-fast03.ps)

Why does file system prefetching work? - Shriver, Small, Smith (1999) (Correct)  
tree structure on disk the root of the tree is an **inode**. The **inode** contains the disk addresses to the  
Why does file system **prefetching** work? Elizabeth Shriver Information Sciences  
Why does file system **prefetching** work? Elizabeth Shriver Information Sciences  
[www.bell-labs.com/user/shriver/postscript/prefetching-usenix99.ps](http://www.bell-labs.com/user/shriver/postscript/prefetching-usenix99.ps)

Persistent Store In A Dynamic Resource Management Environment - Bridgland (1994) (Correct)  
**prefetching** :76 4.1 The **inode** structure for an object comprising 6 blocks :  
: 33 2.4.3 **Prefetching** :  
PSM which implements caching, memory mapping and **prefetching** :76 4.1 The **inode**  
[ftp.cs.man.ac.uk/pub/cnc/j.bridgland.msc.thesis.ps.gz](http://ftp.cs.man.ac.uk/pub/cnc/j.bridgland.msc.thesis.ps.gz)

CLFS Design: A Parallel File Manager for Multicomputers - Perez, Carretero, de.. (1994) (Correct)  
: 10 2.2.3 **Inode** Organization :  
: 37 5.2.2 **Prefetching** Algorithm :  
section: ffl Cache policy and management ffl **Prefetching** algorithm ffl Consistency algorithm All those  
[laurel.datsi.fi.upm.es/~gp/publications/datsi82.1.ps.Z](http://laurel.datsi.fi.upm.es/~gp/publications/datsi82.1.ps.Z)

LFS Design: A Parallel File Server for Multicomputers - Carretero, Pérez.. (1994) (Correct)  
: 11 2.2.3 **Inode** Organization :  
: 38 5.2.2 **Prefetching** Algorithm  
:38 5.2.2 **Prefetching** Algorithm  
[laurel.datsi.fi.upm.es/~gp/publications/datsi81.1.ps.Z](http://laurel.datsi.fi.upm.es/~gp/publications/datsi81.1.ps.Z)

Try your query at: [Amazon](#) [Barnes & Noble](#) [Google \(RI\)](#) [Google \(Web\)](#) [CSB](#) [DBLP](#)

CiteSeer - [citeseer.org](http://citeseer.org) - [Terms of Service](#) - [Privacy Policy](#) - Copyright © 1997-2002 [NEC Research Institute](#)



An additional order for non removable disk is thus generated, which causes again Seek delays. These do not lead however to so large time delays for the processes, as they become necessary without Prefetching. The Inodes is asynchronously ahead-read, since their contents are not needed for the time of reading in yet. In addition the Inode consequences, which are needed for locating the data blocks, are read in not at one time but gradually. If this would not be the case, then the process would have to wait, until all Inodes is loaded from the root to the interesting sheet of the Inode tree successively and evaluated, until the data block is requested.